

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHÓA KHOA HỌC MÁY TÍNH**

----//----



**BÁO CÁO ĐỒ ÁN CUỐI KỲ  
MÔN: TRUY VẤN THÔNG TIN ĐA PHƯƠNG TIỆN  
CS336.J21  
ĐỀ TÀI: TRUY VẤN CƠ SỞ DỮ LIỆU VỚI ẢNH  
MẪU**

Giảng viên hướng dẫn: **TS. NGUYỄN VINH TIỆP**

Sinh viên thực hiện: **Nguyễn Xuân Duy Hiễn – 16521670**

**Trần Phước Lợi – 16521723**

**Nguyễn Nhật Trương – 16521336**

*TP.Hồ Chí Minh, tháng 6 năm 2019*

## MỤC LỤC

|   |    |
|---|----|
| 1. Tổng quan đề tài .....   | 4  |
| 1.1. Đặt vấn đề .....   | 4  |
| 1.2. Mục tiêu nghiên cứu .....  | 4  |
| 1.3. Nội dung thực hiện .....   | 4  |
| 1.4. Đề tài kế thừa .....   | 4  |
| 2. Cơ sở lý thuyết .....  | 4  |
| 2.1. Rút trích đặc trưng SIFT (Scale-Invariant Feature Transform) .....           | 4  |
| a. Scale-space extrema detection .....  | 4  |
| b. Keypoint localization .....  | 4  |
| c. Orientation Assignment .....   | 4  |
| d. Keypoint descriptor .....  | 4  |
| 2.2. Cải tiến rút trích đặc trưng ROOTSIFT .....                                  | 4  |
| 2.3. Thuật toán K-Means .....   | 4  |
| 2.4. Mô hình Bag of Visual Word .....   | 4  |
| 2.5 Các bước thực hiện .....  | 4  |
| 3.1. Phân tích bài toán. ....   | 5  |
| 3.2. Phân tích cấu trúc dữ liệu. ....   | 5  |
| 3.3. Tìm hiểu giải thuật .....  | 5  |
| 3.3.1. Các bước giải thuật .....  | 5  |
| 3.3.2. Vấn đề 1: Rút trích đặc trưng SIFT của từng ảnh. ....                      | 6  |
| 3.3.3. Vấn đề 2: Rút trích đặc trưng rootSIFT của từng ảnh. ....                  | 6  |
| 3.3.5. Vấn đề 3: Đưa ảnh về dạng vector (sparse vector) .....                     | 6  |
| 3.3.6. Vấn đề 4: Đánh chỉ mục. ....   | 6  |
| 3.3.7. Vấn đề 5: Truy vấn ảnh. ....   | 6  |
| 4. Chạy chương trình .....  | 7  |
| 5. Cải tiến truy vấn. ....  | 10 |
| 5.1. Mở rộng câu truy vấn riêng biệt – Discriminative Query Expansion (DQE) ..... | 10 |
| 5.2. Database-side feature augmentation .....                                     | 12 |
| 6. Phân công .....  | 13 |

## *Lời mở đầu*

*Môn học Truy vấn đa phương tiện cung cấp cho chúng em những kiến thức cơ bản về ngôn ngữ lập trình python, matlab..., những nguyên lý, phương pháp và kỹ thuật xử lý ảnh cũng như văn bản đa phương tiện. Đồng thời cung cấp kiến thức giúp chúng em hiểu rõ các cơ chế xử lý của các thành phần về lĩnh vực đa phương tiện*

*Đề tài Truy vấn cơ sở dữ liệu với ảnh mẫu là một chủ đề quen thuộc với nhiều sinh viên IT, để hiểu rõ hơn cơ chế của phân tích ảnh chúng em quyết định tìm hiểu sâu về truy vấn ảnh cũng như cơ chế hoạt động bên trong đó cùng với các vấn đề liên quan. Đây cũng là một chủ đề chúng em rất thích, chính vì vậy chúng em đã chọn đề tài này cho đồ án cuối kỳ này.*

*Trong quá trình hoàn thành đề tài, chúng em đã tìm hiểu được các khái niệm về biến nói chung và một số khái niệm, đặc điểm về biến trong các ngôn ngữ lập trình phổ biến như: python, matlab...vv cũng như kỹ năng tìm hiểu tham khảo tài liệu và các làm việc nhóm thuyết trình.*

*Chúng em xin trân trọng cảm ơn sự hướng dẫn tận tình của thầy Nguyễn Vinh Tiếp về mặt chuyên môn cũng như định hướng. Vì kiến thức còn hạn hẹp nên trong quá trình thực hiện đề tài không thể tránh khỏi sai sót. Vì vậy rất mong nhận được sự góp ý của thầy để đề tài có thể hoàn thiện hơn nữa.*

## **1. Tổng quan đề tài**

### **1.1. Đặt vấn đề**

### **1.2. Mục tiêu nghiên cứu**

### **1.3. Nội dung thực hiện**

### **1.4 Đề tài kế thừa**

## **2. Cơ sở lý thuyết**

### **2.1. Rút trích đặc trưng SIFT (Scale-Invariant Feature Transform)**

### **2.2. Cải tiến rút trích đặc trưng ROOTSIFT**

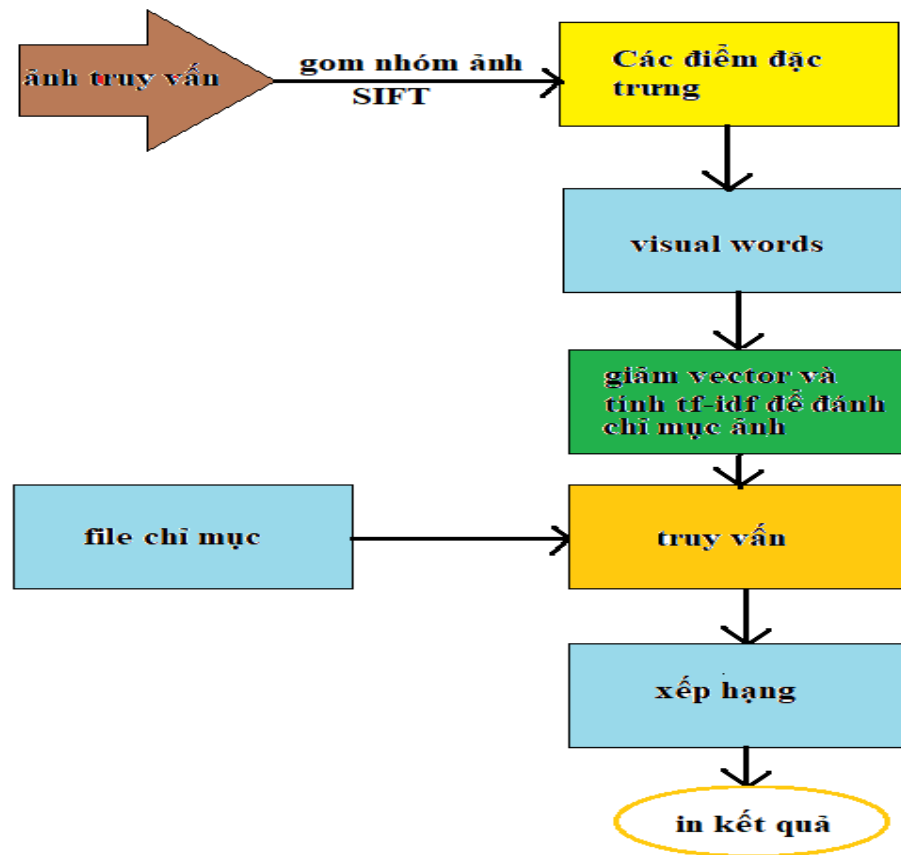
### **2.3. Thuật toán K-Means**

### **2.4. Mô hình Bag of Visual Word**

### **2.5 Các bước thực hiện**

## **3. Phân tích hệ thống truy vấn ảnh mẫu**

### 3.1. Phân tích bài toán.



Mỗi khi phát sinh một yêu cầu truy vấn ảnh thì hệ thống chỉ rút trích các đặc trưng của ảnh truy vấn và bắt đầu so sánh tỉ lệ với các vector được rút trích từ các ảnh mẫu. Hệ thống truy vấn này để xây dựng ứng dụng truy vấn ảnh bằng sử dụng thuật toán Bag of visual words để xử lý đưa hình ảnh về dạng vector so sánh sử dụng if-idf để xác định mức độ tương đồng của các vector khi so sánh.

### 3.2. Phân tích cấu trúc dữ liệu.

### 3.3. Tìm hiểu giải thuật.

#### 3.3.1. Các bước giải thuật.

### **3.3.2. Vấn đề 1: Rút trích đặc trưng SIFT của từng ảnh.**

### **3.3.3. Vấn đề 2: Rút trích đặc trưng rootSIFT của từng ảnh.**

#### **Giải thuật của RootSIFT như sau:**

- Trích xuất SIFT Feature Vectors cho tất cả keypoints, chẳng hạn như sử dụng function `SIFT_create()` của OpenCV;
- Thực hiện L1-normalizing cho từng SIFT Feature Vector;
- Tính căn bậc hai cho từng phần tử trong SIFT Feature Vector đó.

### **3.3.4. Vấn đề 2: Xây dựng từ điển.**

Sử dụng hàm `ccvBowGetDict` để lấy thông tin đặc trưng trong mảng “features”, sử dụng phương pháp gom nhóm k-mean và dựa vào đó, trả về words (1 từ là 1 cluster), danh sách các word này chính là từ điển, từ điển được lưu vào biến “dict\_words”

### **3.3.5. Vấn đề 3: Đưa ảnh về dạng vector (sparse vector).**

Sử dụng hàm `ccvBowGetWords` để đưa thông tin các đặc trưng có trong mảng features về vector, mỗi vector tương ứng với 1 phần tử trong mảng “words”.

### **3.3.6. Vấn đề 4: Đánh chỉ mục.**

Sử dụng hàm `ccvInvFileInsert` để thực hiện thao tác đánh chỉ mục.

Sử dụng hàm `ccvInvFileCompStats` để tính toán những chỉ số cần thiết cho việc tìm kiếm. Với một số điều kiện: sử dụng tf-idf weighting, chuẩn hóa các giá trị trong vector để tổng histogram = 1.

### **3.3.7. Vấn đề 5: Truy vấn ảnh.**

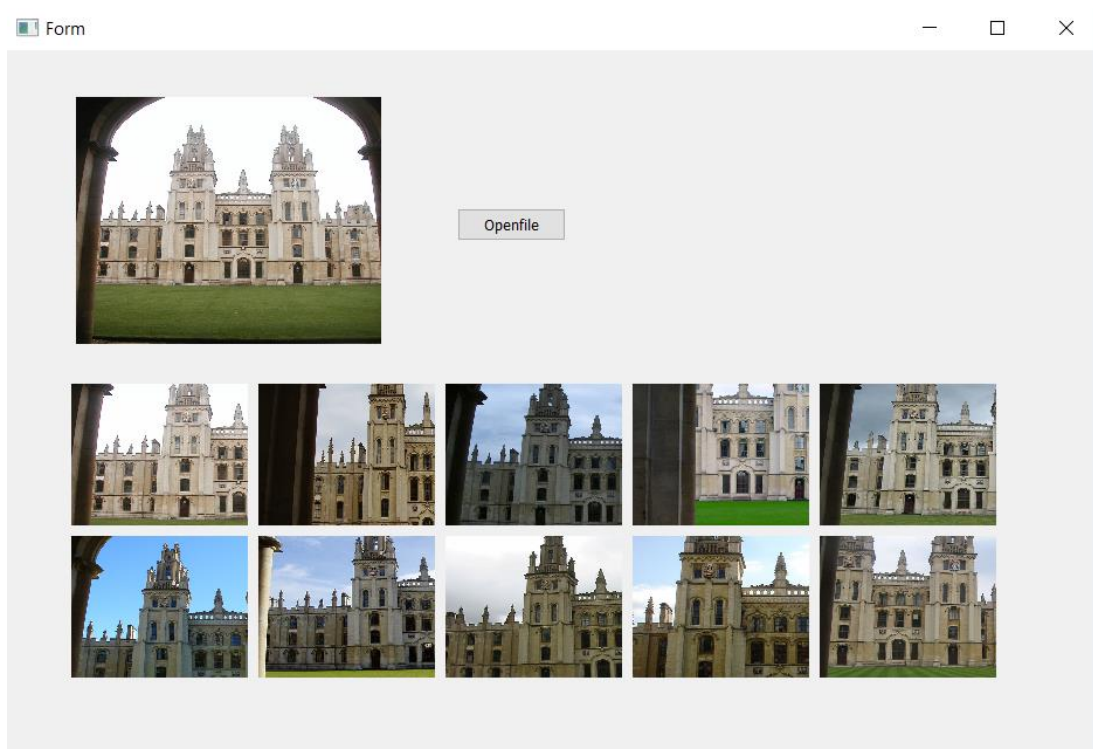
Sau khi đã rút trích đặc trưng của ảnh truy vấn và đưa ảnh truy vấn về vector. Sử dụng hàm `ccvInvFileSearch` để tiến hành truy vấn, với độ đo được sử dụng là Euclidean l1, chuẩn hóa các giá trị trong vector để tổng histogram = 1. Kết quả trả về sẽ là những ảnh gần giống với ảnh truy vấn nhất cùng với độ đo khoảng cách tương ứng.

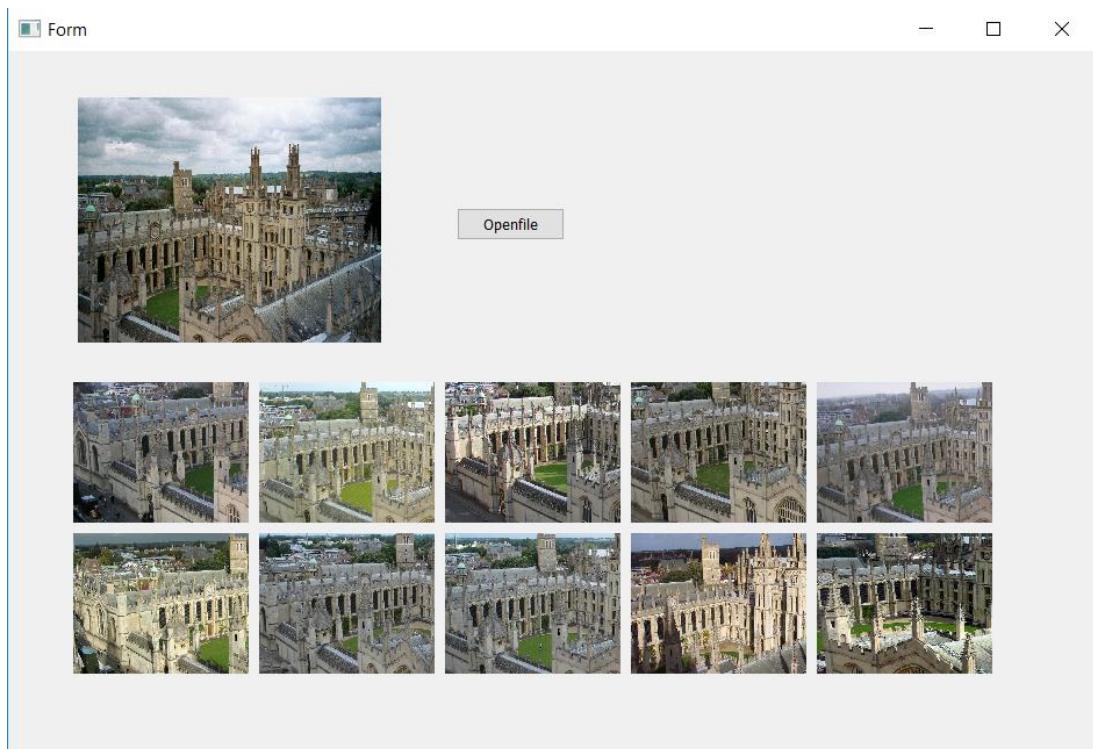
Danh sách ảnh kết quả từ liên quan nhất đến ít liên quan nhất được lưu vào biến “ids”, cùng với khoảng cách tương ứng giữa ảnh kết quả và ảnh truy vấn được lưu vào biến “dists”.

#### 4. Chạy chương trình

Sau khi chạy chương trình, thực hiện trên ngôn ngữ lập trình python

Kết quả của chương trình đa phần cho ra kết quả ảnh mẫu tương tự với ảnh truy vấn ban đầu tuy nhiên vẫn còn nhiều bức hình vẫn chưa ra kết quả đúng chẳng hạn như bức hình dưới:





Hình truy vấn: ashmolean\_00058, ảnh mẫu bị sai: oxford\_01751.jpg, balliol\_00090.jpg,...

Ưu và nhược điểm

- Ưu điểm: Các kết quả đúng với đa phần ảnh, ít tốn không gian lưu trữ cũng như đánh chỉ mục, tốc độ là nhanh hơn so với các phương pháp lưu trữ vector thông thường
- Nhược điểm: Chưa thực sự chạy nhanh với tập dữ liệu lớn cũng như một số kết quả còn sai, độ chính xác còn thấp (precision)

| STT | Query image      | Accuracy |
|-----|------------------|----------|
| 1   | all_souls_000013 | 1.0      |
| 2   | all_souls_000026 | 0,8946   |
| 3   | all_souls_000051 | 0.7536   |
| 4   | ashmolean_000000 | 0.1643   |
| 5   | ashmolean_000007 | 0.2972   |
| 6   | ashmolean_000058 | 0.7477   |
| 7   | ashmolean_000269 | 0.9666   |
| 8   | ashmolean_000305 | 0.125    |
| 9   | balliol_000051   | 0.8333   |



|    |                      |        |
|----|----------------------|--------|
| 10 | balliol_000167       | 0.6791 |
| 11 | balliol_000187       | 1.0    |
| 12 | balliol_000194       | 0.3333 |
| 13 | bodleian_000107      | 0      |
| 14 | bodleian_000108      | 0.8333 |
| 15 | bodleian_000132      | 0.3666 |
| 16 | bodleian_000163      | 0.8055 |
| 17 | bodleian_000407      | 0.1111 |
| 18 | christ_church_000179 | 0.875  |
| 19 | christ_church_000999 | 0.7876 |
| 20 | christ_church_001020 | 0.6166 |
| 21 | cornmarket_000019    | 0.8515 |
| 22 | cornmarket_000047    | 0.7809 |
| 23 | cornmarket_000105    | 0.4655 |
| 24 | cornmarket_000131    | 0.5879 |
| 25 | hertford_000015      | 0      |
| 26 | hertford_000027      | 1.0    |
| 27 | hertford_000063      | 0      |
| 28 | keble_000028         | 1.0    |
| 29 | keble_000055         | 1.0    |
| 30 | keble_000214         | 0.95   |
| 31 | keble_000227         | 0.8556 |
| 32 | keble_000245         | 0.8894 |
| 33 | magdalen_000058      | 0.8133 |
| 34 | magdalen_000078      | 0.9022 |
| 35 | magdalen_000560      | 0.6485 |
| 36 | oxford_000317        | 1.0    |
| 37 | oxford_000545        | 1.0    |
| 38 | oxford_001115        | 0.9166 |
| 39 | oxford_001752        | 0.7    |
| 40 | oxford_001753        | 1.0    |
| 41 | oxford_002416        | 1.0    |
| 42 | oxford_002562        | 0      |
| 43 | oxford_002734        | 0.9166 |
| 44 | oxford_002904        | 0.125  |
| 45 | oxford_002985        | 1.0    |
| 46 | oxford_003335        | 0.8333 |
| 47 | oxford_003410        | 1.0    |
| 48 | pitt_rivers_000033   | 1.0    |
| 49 | pitt_rivers_000058   | 0.5    |
| 50 | pitt_rivers_000087   | 0.5833 |
| 51 | pitt_rivers_000119   | 0.8512 |
| 52 | pitt_rivers_000153   | 0.5420 |

|    |                         |        |
|----|-------------------------|--------|
| 53 | radcliffe_camera_000095 | 0.8469 |
| 54 | radcliffe_camera_000519 | 0.3840 |
| 55 | radcliffe_camera_000523 | 0.7807 |

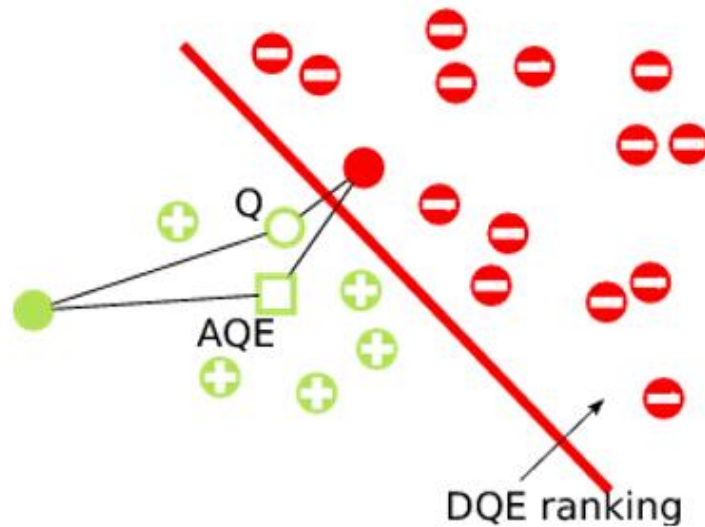
## 5. Cải tiến truy vấn.

Mục tiêu của cải tiến truy vấn là để tăng tốc độ truy vấn bằng hình ảnh của mô hình này với tập dữ liệu với kích thước lớn hoặc rất lớn. Có 2 phương pháp cải tiến truy vấn (improve retrieval) đó là:

- Phương pháp mở rộng câu truy vấn riêng biệt
- Database-side feature augmentation (sử dụng đồ thị)

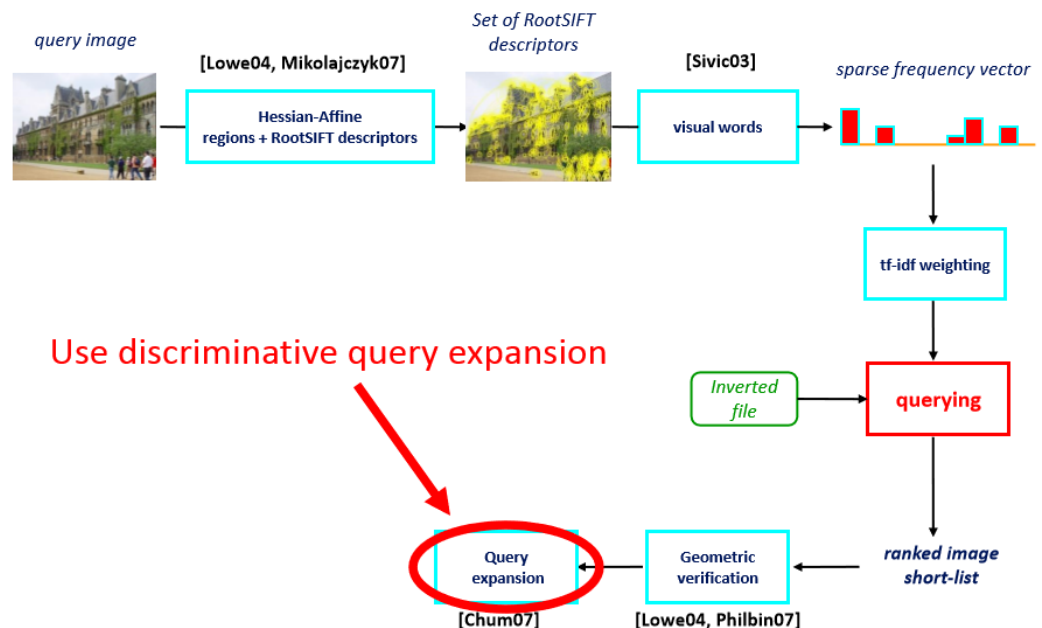
### 5.1. Mở rộng câu truy vấn riêng biệt – Discriminative Query Expansion (DQE)

- Các vector của Bag of Visual Words trong không gian được xây dựng nên mô hình truy vấn đa dạng hơn (riêng biệt)
- Average Query Expansion (AQE)
  - + Được tính bằng cách lấy trung bình của các vector Bag of Visual Word và truy vấn lại
  - + Đặc điểm là tốc độ cũng tương tự nhưng 1 số trường hợp còn chạy nhanh hơn
- Để mở rộng câu truy vấn riêng biệt cần thực hiện như sau:
  - Sử dụng câu truy vấn mở rộng của vector Bag of Visual Words vừa tính ở AQE gán nhãn là positive
  - Đối với những bức ảnh có rank thấp thì sẽ được gán là negative
  - Tính khoảng từ vector đến các ảnh, sau đó chia ra 2 phần được ngăn cách bởi đường decision boundary



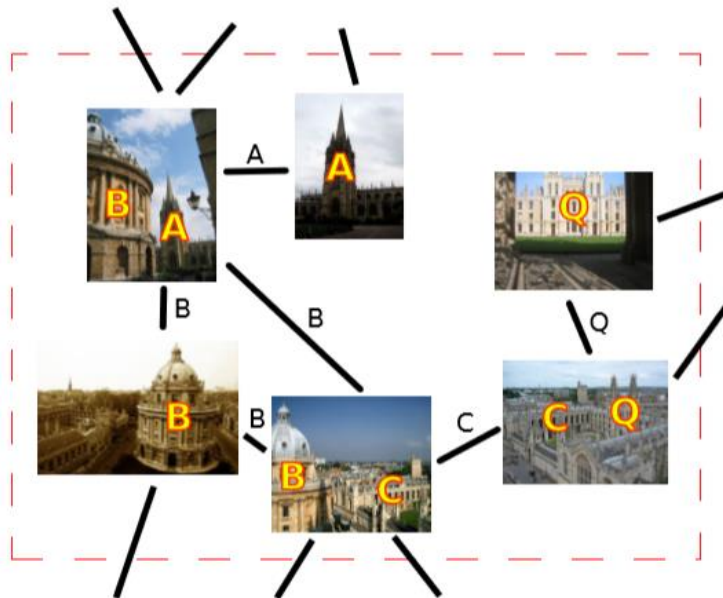
### Quá trình gán nhãn cho các ảnh training

- Sắp hạng lại các ảnh theo cấu trúc chỉ mục ngược
- Đặc điểm của mô hình này là Average Query Expansion thì vector BoW có trọng số trung bình, còn đối với Discriminative Query Expansion thì các vector đã được training dữ liệu và khoảng thời gian tính toán là không đáng kể
- Mô hình này gia tăng đáng kể về hiệu năng cũng như không tốn thêm chi phí



## 5.2. Database-side feature augmentation

- Mô hình này được biểu diễn dưới dạng đồ thị
  - Với Đỉnh: hình ảnh
  - Cạnh sẽ nối các hình ảnh với các Object giống nhau lại
- Tính toán trên đồ thị thông qua hệ thống truy vấn trên từng ảnh cơ sở và lần lượt lưu lại các hình ảnh trong không gian
- Ở đây mỗi hình sẽ được tăng cường(Augment) bởi các ảnh lân cận
- Sắp xếp lại các hình theo thứ tự
- Độ hiệu quả là 28% so với phương pháp gốc, cải thiện được chỉ số Recall tuy nhiên Precision bị giảm đi



*Đồ thị ảnh*

## 6. Phân công

| STT | Tên         | Công việc được giao         | Mức độ hoàn thành (%) |
|-----|-------------|-----------------------------|-----------------------|
| 1   | Hiền Trương | Báo cáo phần 4.3, 4.4       | 100%                  |
| 2   | Lợi Trương  | Báo cáo phần 3, 5, 6        | 100%                  |
| 3   | Trương      | Báo cáo phần 4.1 , 4.2, 4.4 | 100%                  |

## 7. Tài liệu tham khảo

- Bài báo “Three things everyone should know to improve object retrieval” R.Arandjelovic:  
<https://www.robots.ox.ac.uk/~vgg/publications/2012/Arandjelovic12/arandjelovic12.pdf>
- Đặc trưng SIFT:  
<https://viblo.asia/p/gioi-thieu-ve-scale-invariant-feature-transform-z3NVRkoLR9xn>
- K-means clustering:  
[https://en.wikipedia.org/wiki/K-means\\_clustering](https://en.wikipedia.org/wiki/K-means_clustering)
- Bag of Visual Words:  
[http://www.robots.ox.ac.uk/~az/icvss08\\_az\\_bow.pdf](http://www.robots.ox.ac.uk/~az/icvss08_az_bow.pdf)
- <https://github.com/mohamedadaly/caltech-image-search>
- <https://quyv.wordpress.com/2016/07/24/bag-of-word-model/>
- <https://machinelearningcoban.com/2017/01/01/kmeans/>
- <https://kipalog.com/posts/Gioi-thieu-giai-thuat-SIFT-de-nhan-dang-anh>
- <https://viblo.asia/p/gioi-thieu-ve-scale-invariant-feature-transform-z3NVRkoLR9xn>
- <https://machinelearningcoban.com/2017/01/01/kmeans/#tom-tat-thuat-toan>

**Nhận xét của giảng viên hướng dẫn:**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....